AssetModel.js

Die Datei assetModel.js definiert die **Struktur der Asset-Datenbanktabelle** in MongoDB.  
Jedes Asset hat eine **eindeutige ID (UUID)** und Eigenschaften wie Name, Typ, Standort und Status.

**Ablauf des Codes (assetModel.js)**

1️. **Import von benötigten Modulen**

import mongoose from "mongoose";

import { v4 as uuidv4 } from "uuid"; // UUID-Generator importieren

→ **mongoose** wird für die Verbindung zur MongoDB genutzt.  
→ **uuidv4** wird verwendet, um **automatisch eine eindeutige ID (assetId)** zu generieren.

2️. **Erstellen des Asset-Schemas**

const assetSchema = new mongoose.Schema({

assetId: { type: String, default: uuidv4, unique: true }, // Automatische UUID

name: { type: String, required: true }, // Name des Assets

type: { type: String, required: true }, // Typ des Assets (z. B. "Elektronik")

location: { type: String, required: true }, // Standort des Assets

status: { type: String, default: "In Betrieb" }, // Standardstatus "In Betrieb"

assignedTo: { type: String, default: "Unassigned" } // Standardmäßig nicht zugewiesen

},

{ timestamps: true }); // Automatische Erstellung von "createdAt" und "updatedAt"

→ Definiert, welche **Eigenschaften jedes Asset haben muss**.  
→ **timestamps: true** sorgt dafür, dass createdAt und updatedAt automatisch gesetzt werden.

3️. **Erstellen des Asset-Modells**

const Asset = mongoose.model("Asset", assetSchema);

export default Asset;

→ Erstellt ein **MongoDB-Modell Asset** basierend auf dem Schema.  
→ Wird später in assetController.js verwendet, um mit der Datenbank zu arbeiten.

**📊 Eigenschaften eines Assets**

|  |  |  |
| --- | --- | --- |
| **Feld** | **Typ** | **Beschreibung** |
| assetId | String | Automatisch generierte **UUID** |
| name | String | Name des Assets (Pflichtfeld) |
| type | String | Art des Assets, z. B. "Elektronik" |
| location | String | Standort, z. B. "Büro" |
| status | String | Standardmäßig "In Betrieb" |
| assignedTo | String | Standardmäßig "Unassigned" |
| createdAt | Date | Automatisch generiertes Erstellungsdatum |
| updatedAt | Date | Automatisch generiertes Aktualisierungsdatum |

**🛠️ Warum ist dieser Code wichtig?**

✅ Jedes Asset hat **eine eindeutige ID (UUID)**, was Fehler mit Duplikaten vermeidet.  
✅ Die Datenstruktur ist **klar definiert**, was die Konsistenz in der Datenbank sicherstellt.  
✅ Automatische **Zeitstempel** helfen dabei, Änderungen nachzuverfolgen.

**📌 Fazit:**  
Die Datei assetModel.js ist **die Grundlage der Datenbankstruktur**. Sie stellt sicher, dass jedes Asset die richtigen Eigenschaften hat und in der MongoDB-Datenbank richtig gespeichert wird. 🚀